Nota de Aplicação NAP103

Configuração de Redes Ethernet

Sumário

1. Introdução

Este documento tem como objetivo orientar desenvolvedores de sistemas em arquiteturas que envolvam redes Ethernet TCP/IP.

São apresentadas diretrizes de como configurar os diferentes elementos da rede, entre eles, CP, driver de supervisão e microcomputador, bem como orientações de uso de ferramentas de auxílio a manutenção da rede.

Aspectos relacionados à performance do sistema são abordados, buscando a melhor configuração do sistema para atingir o melhor desempenho.

O capítulo a seguir apresenta informações preliminares para o bom entendimento desta nota de aplicação.

2. Informações Preliminares

2.1 Transação

Uma transação de comunicação é composta por uma seqüência de duas mensagens: requisição e resposta. A requisição pode ser interpretada como um comando, e a resposta como o resultado da execução do comando.

Estas mensagens são trocadas, normalmente, entre dois nós. O objetivo de uma transação é a execução de um serviço de rede. Existe um nó que requisita o serviço (mestre ou cliente), e outro que fornece o serviço (escravo ou servidor). Inicialmente, o cliente envia uma mensagem (requisição) para o servidor. A seguir o servidor envia uma mensagem para o cliente (resposta).

Na requisição, o cliente pode informar para o servidor:

- o tipo de serviço que deve ser executado;
- pacote de dados que devem ser recebidos pelo servidor (exemplo: o cliente deseja escrever variáveis no servidor).

Na resposta, o servidor pode informar para o cliente:

- que o serviço solicitado foi reconhecido e executado;
- que o serviço solicitado não foi reconhecido, ou que não foi executado por algum motivo;
- pacote de dados requisitado pelo cliente (exemplo: o cliente solicitou variáveis do servidor).

Uma transação é finalizada quando o nó requisitor recebe:

- uma mensagem de resposta sem dados, confirmando o serviço, para o caso de serviços sem troca de dados;

- uma mensagem de resposta com dados, confirmando o serviço, para o caso de serviços com um único pacote de dados;
- a última mensagem de resposta com dados, confirmando o serviço, para o caso de serviços com múltiplos pacotes de dados;
- por time-out, quando nenhuma mensagem de resposta é recebido pelo requisitor dentro do tempo esperado.

2.2 Protocolo de Comunicação

Uma arquitetura envolvendo comunicação Ethernet TCP/IP pode possuir diferentes protocolos no seu nível de aplicação. Assim sendo, fala-se em Ethernet para o nível de rede, mas em protocolo ALNET II ou MODBUS TCP para o nível de aplicação.

A maioria das interfaces de comunicação baseadas em Ethernet, implementam apenas cinco das sete camadas definidas pelo modelo ISO-OSI:

- 10/100Base-TX: camada física;
- Ethernet: camada de enlace;
- Internet Protocol (IP): camada de rede;
- Transport Control Protocol (TCP): camada de transporte;
- ALNET II ou MODBUS TCP: camada de aplicação.

2.3 Drivers de Comunicação

Drivers de comunicação são softwares responsáveis pela aquisição e envio de dados para dispositivos diversos, e portanto, em diferentes protocolos. Assim, cada vez mais o uso de drivers de comunicação padrão OPC Server são utilizados para comunicação com diferentes dispositivos, como controladores programáveis, minimizando a necessidade de novos desenvolvimentos de drivers para diferentes aplicações de supervisão e controle de processos.

Um OPC (**O**LE [**O**bject **L**inking and **E**mbedding] for **P**rocess **C**ontrol) Server nada mais é do que um servidor para clientes OPC, que torna possível a troca de dados entre dispositivos utilizando protocolos de aplicação diferentes.

O computador, ou estação, se comporta como um nó da rede. O driver de comunicação de um sistema de supervisão deve disponibilizar bases de dados similares as dos CPs com os quais se comunica (áreas de memória para dados e serviços).

A aquisição de dados, feita pelo driver de comunicação de um sistema de supervisão, pode se dar através das seguintes maneiras:

- aquisição cíclica de dados: o driver de comunicação busca a informação em um CP, e a escreve na base de dados associada a este CP;
- recepção de mensagens não solicitadas: os CPs escrevem em uma base de dados dedicada do driver da estação de supervisão, enviando requisições para o endereço do computador na rede.

No primeiro tem-se o driver operando como dispositivo cliente, enquanto que no segundo, como servidor. Nada impede porém, do driver operar das duas maneiras: como cliente, requisitando dados de uma forma geral, e como servidor, recebendo informações de alarmes por exemplo.

3. Princípios de Funcionamento

O protocolo ALNET II é utilizado na camada de aplicação dos drivers de comunicação de sistemas de supervisão que rodam sobre o protocolo TCP/IP através de uma rede Ethernet, e que desejam se comunicar com CPs Altus através de alguma das interfaces de comunicação a seguir relacionadas:

Código	Denominação
AL-3405	Interface para Rede Ethernet
AL-3412	Interface Ethernet 10/100 Mbits/s
AL-3414	Interface Ethernet Redundante Modbus TCP
PO7091	Interface Ethernet Industrial
PO7092	Interface Ethernet Industrial 10-100Mbps
PO9900	Webgate Interface Ethernet com WEB Server
PO9901	Webgate Plus Ethernet com WEB Server

Interfaces Altus com suporte ao protocolo de aplicação ALNET II

O protocolo MODBUS TCP, ou MODBUS RTU sobre TCP/IP, também pode ser utilizado pela comada de aplicação dos drivers de comunicação para se comunicar com CPs Altus através da seguinte interface de comunicação, quando configurada para tal:

Código	Denominação
AL-3414	Interface Ethernet Redundante Modbus TCP

Interfaces Altus com suporte ao protocolo de aplicação MODBUS TCP

3.1 Limites de Dados por Transação ALNET II

Comunicações com os CPs Altus podem ser realizadas através de comandos de leitura ou escrita de blocos de dados limitados em 220 bytes de dados. Determinados drivers de comunicação podem executar automaticamente o agrupamento de operandos para aproveitar todo o limite do bloco. Outros não, devendo ficar a cargo do usuário o agrupamento dos dados de forma manual.

No entanto, para qualquer driver, o limite para estas transações podem variar caso se esteja utilizando os dispositivos WebGate ou WebGate Plus – PO9900 ou PO9901 – uma vez que para este, os blocos de dados devem ser definidos com até 128 bytes.

O detalhamento do número máximo de operandos que podem ser lidos ou escritos em cada bloco de dados, utilizando-se as interfaces Ethernet Altus é apresentado na tabela a seguir:

Tipo de Operando	N° de Bytes por Operando	N° Máximo de operandos por bloco para AL-3405, AL-3412, AL-3414, PO7091 e PO7092	N° Máximo de operandos por bloco para PO9900 e PO9901
Octetos de E/S	1 byte	220 operandos	64 operandos
Auxiliar	1 byte	220 operandos	64 operandos
Memória	2 bytes	110 operandos	64 operandos
Decimal, Real e Inteiro	4 bytes	55 operandos	32 operandos
Tabela de Memória	2 bytes por posição	110 posições	64 posições
Tabela de Decimal, Tabela de Real e Tabela de Inteiro	4 bytes por posição	55 posições	32 posições

Número máximo de operandos por bloco de dados

A configuração para leitura de operandos do CP em blocos de dados deve observar os limites para cada tipo de operando. Caso múltiplos deste limite sejam ultrapassados, uma nova requisição é gerada para o CP. Se isto for necessário, é indicado que se utilize todo o range de operandos do novo bloco.

Exemplo de configuração, considerando os limites da interface AL-3412:

Operandos Monitorados	N° de Bytes Solicitados	N° Máximo de bytes por bloco de dados	N° de Requisições Efetuadas ao CP
M0000 à M0099	200	220	1
M1000 à M1114	230	220	2
M2000 à M2549	1100	220	5
M3000 à M3029	60	220	1
M3060 à M3089	60	220	1
A0100 à A0299	200	220	1
I0200 à I0299	400	220	2
F0300 à M0461	648	220	3

Exemplo de configuração de leitura de operandos em blocos de dados

No exemplo de configuração acima, podemos observar duas situações de má configuração:

- o bloco de leitura do operando M1000 à M1114, irá exigir do driver a geração de duas mensagens de requisição, sendo que na segunda serão requisitados apenas cinco operandos memória, ou seja, 10 bytes de dados.

  O usuário deve estar ter em mente que o tempo necessário para leitura de um pacote com 10 bytes de dados é praticamente o mesmo que o tempo para leitura de um pacote com 220 bytes de dados. Isto sugere que os blocos configurados sejam sempre múltiplos do limite de bytes por bloco de dados suportados pela interface utilizada.

- o bloco de leitura do operando M3000 à M3029 e o bloco de leitura do operando M3060 ao operando M3089 podem exigir do driver a geração de duas mensagens de requisição. Mesmo que os dados referentes aso operandos M3030 à M3059 não sejam utilizados, sugere-se a configuração de um único bloco, do operando M3000 ao M3089, resultando num bloco com 180 bytes de dados, possível de ser lido através de uma única mensagem de requisição.

  O agrupamento de dados numa única mensagem de requisição, em operandos consecutivos, é um dos requisitos fundamentais para uma boa performance de comunicação.

3.2 Limites de Dados por Transação MODBUS TCP

Os limites dos blocos de dados para transações MODBUS TCP ou MODBUS RTU sobre TCP/IP, utilizando a interface de comunicação AL-3414, estão de acordo com a norma que define o protocolo MODBUS, e podem ser encontradas no manual de utilização da interface AL-3414.

3.3 Tratamento das Requisições pelo CP

Uma característica intrínseca aos CPs é o fato deles tratarem as mensagens de comunicação, sejam elas provenientes de canais seriais tipo RS232 ou RS485 ou ainda de interfaces de rede Ethernet, apenas ao final de um ciclo de execução. Apesar de ser uma limitação, esta é uma garantia de que os

operandos utilizados pela aplicação não vão mudar de valor durante a execução de um ciclo, o que poderia resultar em resultados desastrosos em determinadas situações.

Esta característica limita as comunicações que ocorrem de forma serial, onde uma segunda requisição só é transmitida pelo cliente/mestre depois que a resposta da primeira requisição for recebida pelo mesmo, o que limita o desempenho da comunicação com o cliente/mestre em uma requisição por ciclo.

Para comunicações através de portas seriais do tipo RS232 ou RS485 não existe uma solução rápida e simples para resolver este problema, mesmo porque a comunicação já está limitada a ocorrer com um único dispositivo. Já nos casos de comunicações ALNET II ou MODBUS TCP, pode-se querer comunicar com mais de um dispositivo ao mesmo tempo, pois a rede utilizada suporta esta situação.

Para estes casos foi criada a instrução LAI, podendo ser empregada em aplicações desenvolvidas em diagramas de blocos e reles (LADDER). Com o emprego desta instrução, além de tratar a requisição ALNET II pendente que normalmente seria tratada ao final de um ciclo de execução, o CP tratará uma requisição adicional pendente para cada instrução LAI definida na aplicação.

3.4 Mensagens Não Solicitadas

O uso das mensagens não solicitadas permite melhorar significativamente a performance do sistema de supervisão, pois o mesmo não precisa manter a leitura cíclica das variáveis do CP que não sofrem alteração, ou que sofrem alterações esporadicamente.

A idéia principal por trás das mensagens não solicitadas é a transmissão de um pacote de dados apenas quando os dados sofrerem alteração. Neste caso é o CP que toma a iniciativa de fazer uma escrita no sistema de supervisão, quando necessário, ao invés do sistema de supervisão permanecer lendo os dados periodicamente.

São exemplos típicos de dados que não necessitam ser lidos pelo sistema periodicamente:

- tabelas de alarmes;
- tabelas de eventos.

3.5 Mensagens Multi Request

Mensagens Multi Request são mensagens Ethernet contendo mais de uma requisição na camada de aplicação, ao invés da tradicional única requisição. Através deste mecanismo o cliente é capaz de requisitar paralelamente, através de uma única conexão com o servidor, mais de uma informação (bloco de dados).

Não quer dizer que uma nova mensagem de requisição será transmitida pelo cliente somente depois que todas as requisições pendentes da mensagem anterior sejam respondidas: assim que uma das respostas é recebida, o cliente já pode disparar uma nova requisição, ou seja, normalmente vão existir mais de uma requisição pendentes no dispositivo servidor.

O servidor, ao receber uma mensagem Ethernet contendo mais de uma requisição na camada de aplicação, deve desmembrar estas requisições e responder cada uma delas isoladamente. Se receber as requisições através de mensagens consecutivas, o servidor deve acumular todas elas para que sejam tratadas e respondidas assim que possível (lembrar que o CP trata requisições somente ao final de um ciclo de execução).

Neste modelo, considerando um CP operando como servidor, ao final de um ciclo de execução, em vez de ter apenas uma requisição, O CP terá várias requisições para serem tratadas e respondidas. Esta é uma das formas possíveis de multiplicar a performance da comunicação.

3.6 Múltiplas Conexões

Tipicamente um dispositivo cliente estabelece uma única conexão com um servidor para troca de dados. Considerando a velocidade da rede Ethernet e das interfaces envolvidas, esta única conexão pode ser suficiente para efetuar a troca de uma grande quantidade de dados entre dois dispositivos.

Ocorre que as interfaces Ethernet dependem também do tempo de ciclo dos CPs, uma vez que a troca de dados entre o CP e sua interface ocorre somente ao final de um ciclo de execução, o que acaba limitando a velocidade de comunicação com outros dispositivos da rede Ethernet.

Uma das soluções para melhorar a performance desta comunicação entre o CP e demais dispositivos da rede Ethernet, é o estabelecimento de múltiplas conexões por parte do cliente com o servidor. A partir de então, em vez de fazer uma única requisição e aguardar pela sua resposta e só então fazer a próxima requisição, o cliente dispara uma requisição específica através de cada uma das conexões estabelecidas com o servidor, de forma paralela.

Neste modelo, considerando um CP operando como servidor, ao final de um ciclo de execução, em vez de ter apenas uma requisição, O CP terá várias requisições para serem tratadas e respondidas. Esta é uma das formas possíveis de multiplicar a performance da comunicação.

4. Driver de Comunicação e Interfaces Ethernet Altus

As diversas interfaces Ethernet fabricadas pela Altus possuem características diferentes, e isto implica em desempenho individual diferenciado. O mesmo vale para os diversos drivers de comunicação disponibilizados pela Altus para os sistemas de supervisão.

As seções a seguir apresentam algumas destas características, discriminando o comportamento de cada uma delas. Tais características e comportamentos devem ser considerados pelo usuário para obter o máximo desempenho de comunicação.

4.1 Tempo de Resposta para uma Transação

As informações apresentadas na próxima tabela foram obtidas a partir de experimentos executados em laboratório, considerando uma única conexão entre o sistema de supervisão e a interface de comunicação do CP, e ainda um tempo de ciclo de execução do CP inferior a 1 ms.

Interface	Tempo de Resposta [ms]
AL-3405	60
AL-3405 modo Turbo	17
AL-3412	7
AL-3414	7
PO7091	90*
PO7092	21*
PO9900	120**
PO9901	120**

Tempo de resposta médio

NOTAS:

* O tempo de resposta pode oscilar em virtude das possibilidades de configuração de alguns parâmetros pelo usuário.

** O tempo de reposta pode oscilar em virtude das configurações de baudrate e do tamanho dos blocos de dados.

4.2 Características Exclusivas de Drivers de Comunicação

A tabela a seguir apresenta as principais características dos drivers de comunicação com suporte ao protocolo de aplicação ALNET II. A descrição funcional das características encontra-se na última seção deste capítulo.

Característica	Driver iFix	Driver Intouch	Driver Elipse	Driver OPC AL-2785
Transmissão de mensagens Multi Request	Não	Até 4 mensagens a partir da versão 8.0.2	Não	Não
Recepção de mensagens Multi Request	Não	Não	Não	Não
Abertura de múltiplas conexões	Não	Não	Até 8 conexões	Até 8 conexões
Suporte a mensagens não solicitadas	Sim	Sim	Sim	Sim
Recepção de mensagens (respostas) de tamanho real	Sim	Não	Não	Não
Comunicação com IPs destino vvv.xxx.yyy.zzz, quando zzz maior que 127	Sim	Depende da versão utilizada	Sim	Sim
Porta de conexão Listen (modo servidor)	405	405	Configurável	Configurável
Porta de conexão com o servidor	405	Configurável	Configurável	Configurável
Tamanho máximo do bloco de dados [bytes]	128	Configurável	Configurável	Configurável
Agrupamento automático de operandos simples numa única requisição	Não	Sim	Não	Sim
Suporte a operandos reais (%F e %TF)	Não	A partir da versão 8.0.6	Sim	Sim
Suporte a operandos inteiros (%I e %TI)	Sim	A partir da versão 8.0.5	Sim	Sim
Suporte a leitura de BITs de operandos tabela memória (BIT de %TM)	Não	Sim	Não	Não
Fechamento da conexão para respostas com erro	Sim	Sim	Sim	Sim

Características dos drivers de comunicação para ALNET II

4.3 Características Exclusivas de Interfaces Ethernet Altus

A tabela a seguir apresenta as principais características das interfaces de comunicação Ethernet da Altus, com suporte ao protocolo de aplicação ALNET II. A descrição funcional das características encontra-se na última seção deste capítulo.

Característica	AL-3405	AL-3412	AL-3414	PO7091 / PO7092	PO9900 / PO9901
Transmissão de mensagens Multi Request	Não	Não	Não	Sim	Sim
Recepção de mensagens Multi Request	Configurável via F-ETHER a partir da versão 2.00	Configurável via F-ETHER a partir da versão 1.00	Configurável via módulo C000	Configurável via módulo C000	Configurável via módulo C000
Transmissão de mensagens (respostas) de tamanho real	Configurável via F-ETHER a partir da versão 2.00	Configurável via F-ETHER a partir da versão 1.00	Configurável via módulo C000*	Configurável via módulo C000	Configurável via módulo C000
Comunicação com IPs destino vvv.xxx.yyy.zzz, quando zzz maior que 127	Sim	Sim	Sim	Sim	Sim
Comunicação entre roteadores através de NAT	Somente se o último campo do endereço IP externo (zzz) for igual ao seu.	Somente se o último campo do endereço IP externo (zzz) for igual ao seu.	Somente se o último campo do endereço IP externo (zzz) for igual ao seu.	Somente se o último campo do endereço IP externo (zzz) for igual ao seu.	Somente se o último campo do endereço IP externo (zzz) for igual ao seu.
Porta de conexão Listen (modo servidor)	405	405	405**	405	405
Porta de conexão com o servidor	405	405	405***	405	405
Tamanho máximo do bloco de dados [bytes]	220	220	220****	220	128

Características das interfaces de comunicação para ALNET II

NOTAS:

* Sempre transmite mensagens de tamanho real para o protocolo de aplicação MODBUS TCP ou MODBUS RTU sobre TCP/IP.

** 502 para o protocolo de aplicação MODBUS TCP ou MODBUS RTU sobre TCP/IP.

*** Configurável para o protocolo de aplicação MODBUS TCP ou MODBUS RTU sobre TCP/IP.

**** De acordo com as definições da norma para o protocolo de aplicação MODBUS TCP ou MODBUS RTU sobre TCP/IP.

4.4 Descrição Funcional das Características

A seguir a descrição funcional das principais características que foram especificadas para os drivers e interfaces de comunicação Ethernet.

Característica	Descrição
Transmissão de mensagens Multi Request	Indica se o dispositivo é capaz de transmitir mensagens do tipo Multi Request.
Recepção de mensagens Multi Request	Indica se o dispositivo é capaz de receber e tratar mensagens do tipo Multi Request.
Abertura de múltiplas conexões	Indica se o dispositivo cliente é capaz de estabelecer múltiplas conexões com o servidor, e disparar requisições distintas através de cada uma delas, de forma paralela.
Suporte a mensagens não solicitadas	Indica que o dispositivo, além de operar como cliente, pode operar como servidor, recebendo requisições de outros dispositivos. O emprego desta metodologia, uso de mensagens não solicitadas, é interessante quando os dados não costumam ficar variando, como é o caso de eventos e alarmes.
Transmissão e recepção de mensagens (respostas) de tamanho real	Historicamente, apesar da norma ALNET II não especificar, toda e qualquer mensagem ALNET II transmitida sobre a rede Ethernet pelas interfaces da Altus possuía 256 bytes, independente da quantidade de dados da requisição/resposta. Tamanho real significa que em vez de possuir exatamente 256 bytes, o tamanho da mensagem é ajustado em função da quantidade de dados da requisição/resposta.
Comunicação com IPs destino vvv.xxx.yyy.zzz, quando zzz maior que 127	Além do endereço IP aparecer descrito na camada de rede de todas as mensagens Ethernet, o último campo do endereço IP (zzz) aparece também na camada de aplicação do protocolo ALNET II, só que multiplicado por dois, dividido em parte alta (byte high) e parte baixa (byte low). Alguns dos dispositivos não são capazes de manipular esta parte do endereço quando ela é maior que 127, pois gera um número maior que 255 quando multiplicado por dois e, portanto, impossível de armazenar apenas na parte baixa (byte low), uma vez que a parte alta (byte high) é mantido com um valor constante.
Porta de conexão Listen (modo servidor)	É pela porta Listen que o servidor fica aguardando as solicitações de conexão por parte dos clientes. Normalmente existe uma porta específica para cada protocolo de aplicação.
Porta de conexão com o servidor	Indica com que porta do servidor este dispositivo, quando operando como cliente, vai tentar estabelecer conexão. Trata-se de um número de 16 BITs presente em todas as mensagens Ethernet, que deve conferir com a porta de conexão Listen do servidor. Ao estabelecer uma conexão, o cliente sorteia uma porta local (valor aleatório diferente das portas reservadas e de qualquer outra que já esteja em uso) e tenta estabelecer conexão com o endereço Listen do servidor.
Tamanho máximo do bloco de dados [bytes]	O tamanho máximo de mensagens de aplicação ALNET II e MODBUS TCP está limitado em 256 bytes. Parte desta mensagem é composta por bytes de identificação de endereço dos dispositivos, códigos de comando e até mesmo bytes de checksum. Logo, não são todos os 256 bytes da mensagem que podem ser preenchidos com dados representando os valores de operandos, por exemplo. O tamanho máximo do bloco de dados representa a quantidade real de bytes de dados que uma mensagem pode possuir e ainda ser válida.
Agrupamento automático de operandos simples numa única requisição	Indica se o dispositivo (driver de comunicação) é capaz de montar requisições concatenando dois ou mais blocos de dados próximos, configurados pelo usuário, tentando se aproximar ao máximo do tamanho limite do bloco de dados.
Suporte a operandos reais (%F e %TF)	Indica que o dispositivo suporta pacotes de dados relacionados a operandos simples ou tabelas de operandos reais (%F ou %TF).
Suporte a operandos inteiros (%I e %TI)	Indica que o dispositivo suporta pacotes de dados relacionados a operandos simples ou tabelas de operandos inteiros (%I ou %TI).
Suporte a leitura de BITs de operandos tabela memória (BIT de %TM)	Indica que o dispositivo suporta montar comandos e tratar respostas referentes a leitura de BITs de posição de tabela memória (%TM).
Fechamento da conexão para respostas com erro	Indica se o dispositivo cliente fecha ou não a conexão estabelecida com o servidor, no caso deste último responder com erros na camada de aplicação da mensagem.
Comunicação entre roteadores através de NAT	O serviço NAT dos roteadores permite o estabelecimento de conexão e troca de dados entre dispositivos de duas redes diferentes, por exemplo rede interna e rede externa (Internet).

Descrição funcional das principais características

5. Desempenho da Comunicação

O desempenho de uma arquitetura Ethernet está diretamente relacionada à qualidade da configuração efetuada nos dispositivos envolvidos. Isto inclui tanto a configuração das interfaces de rede, como a configuração dos drivers de comunicação.

Nas seções a seguir seguem importantes dicas de configuração destes dispositivos, buscando sempre um melhor desempenho de comunicação.

5.1 Requisições Simultâneas

Os dispositivos mestres devem ser configurados para enviarem requisições simultâneas para um ou mais escravos, de modo a paralelizar a aquisição de dados, e com isto, minimizar o tempo para obtenção de todos os blocos de dados.

O número de requisições simultâneas deve ser configurada no driver de comunicação. O parâmetro a ser configurado vai depender da técnica empregada pelo driver para enviar requisições simultâneas, discutidas anteriormente: mensagens Multi Request ou múltiplas conexões.

Na prática, este valor deve ser definido em função do desempenho da interface Ethernet empregada e do tempo de ciclo do CP, posto que de nada adianta enviar muitas requisições e causar uma sobrecarga do sistema, levando a perda de mensagens pela interface e posterior aumento do tráfego na rede devido as retransmissões, piorando o seu desempenho em vez de melhorar.

O uso de requisições simultâneas implica no emprego de instruções que habilitem o tratamento de múltiplas requisições em um único ciclo de execução do CP. A instrução a ser empregada nas aplicações é a LAI - Libera Atualização de Imagem dos Operandos.

Para determinar a quantidade ótima de instruções LAIs a serem incluídas na aplicação, deve-se efetuar o seguinte cálculo: quantidade de LAIs = (tempo de ciclo de execução do CP) / (tempo de resposta da interface).

O tempo de ciclo de execução do CP vai depender da aplicação sendo executada, e é um dado que pode ser levantado pelo usuário através do programador MasterTool quando a aplicação já estiver em funcionamento. O tempo de resposta da interface pode ser obtido na tabela ***Tempo de resposta médio*** deste documento.

O emprego de um maior número de instruções LAI não surtirá efeito, porque a interface não terá tempo para enviar todas as mensagens até o próximo fim do ciclo de execução do CP. Pode implicar sim num maior tempo de ciclo de execução do CP (aproximadamente 1ms por instrução LAI).

ATENÇÃO:

Quando utilizada a interface AL-3414, não é necessário o emprego das instruções LAI. As mesmas podem ser substituídas com vantagens pela definição do parâmetro de configuração "*Número de comunicações ALNET II por ciclo do CP*". Consultar o manual de utilização da interface AL-3414 para maiores informações.

As técnicas de requisições simultâneas não são aplicáveis para CPs conectados a rede Ethernet através das interfaces WebGate e WebGate Plus (PO9900 e PO9901), pois o CP e a interface permanecem interligados através de uma interface serial RS232.

Os valores dos operandos do programa aplicativo podem ser modificados durante a execução de uma instrução LAI, pois um dispositivo pode estar solicitando escritas nos mesmos. Deve-se considerar a influência deste fato ao se inserir esta instrução em pontos intermediários do programa aplicativo. Por este motivo, recomenda-se que instruções LAI sejam inseridas na primeira lógica do programa aplicativo principal da UCP (E001).

5.2 Definição dos Blocos de Dados

O desempenho da comunicação está diretamente relacionada com a correta alocação de operandos do CP. Os CPs recebem forçamentos de forma assíncrona, mas são monitorados ciclicamente pelos sistemas de supervisão, assim a maior responsabilidade da definição corresponde a interface de leitura dos CPs.

Por características dos protocolos ALNET I, ALNET II e Ethernet, e da forma como os mesmos são utilizados nas implementações de drivers de sistemas de supervisão e controle, a monitoração de dados através de operandos tabelas, ou através de blocos de operandos, é mais eficiente que a monitoração de operandos simples independentes, na maioria dos casos. O segredo do ganho de desempenho está na forma como os operandos são agrupados numa mesma resposta ou numa mesma mensagem enviada do CP para as estações de supervisão e controle.

ATENÇÃO

Este documento dará ênfase a comunicação através de tabelas, embora o termo tabela possa ser um sinônimo de bloco de operandos simples. O importante é seguir os demais princípios que serão apresentados, independente do tipo de operando utilizado, cuja escolha torna-se uma questão de opção de implementação.

O uso de tabelas possui uma maior flexibilidade de alocação, permitindo o aumento de um bloco lógico sem necessidade de movimentação de operandos e alteração de código. Já um bloco de operandos simples simplifica o aplicativo do CP, eliminando a necessidade das instruções MOT.

Seguem alguns princípios práticos na definição da comunicação e dos blocos de dados. Nenhum deles é absoluto, sendo necessário priorizá-los conforme a situação do projeto:

- As informações devem ser adquiridas do CP através de tabelas exclusivas de leitura, sendo os forçamentos realizados através de tabelas de escrita ou outros operandos simples, tais como auxiliares de comando. Operandos de leitura e escrita também podem ser utilizados para realizar uma troca de sinalização entre CP e estação de supervisão. A escrita de posições de tabelas, memórias ou auxiliares é equivalente no nível de protocolo, sendo aconselhável a utilização de auxiliares para comandos digitais apenas para um melhor aproveitamento dos operandos do CP, que possui um espaço restrito de RAM. Naturalmente, quando o sistema de supervisão permite o forçamento de tabelas inteiras, este recurso também deve ser utilizado para otimizar a escrita de conjuntos de dados.
- Quando o CP precisa notificar a existência de novos dados, sendo necessária a confirmação desta leitura pela estação, pode ser alocada a primeira posição da tabela de monitoração destas informações para realizar a leitura e escrita desta sinalização. Neste caso a tabela continuaria a ser essencialmente uma tabela de monitoração correspondente a TAGs de leitura e apenas o primeiro operando estaria associado a um TAG adicional de escrita ou corresponderia a um TAG de leitura e escrita.
- Sempre que os recursos da estação e os requisitos de tempo do aplicativo do CP permitirem, os dados a serem lidos ou escritos, devem ser compactados nas tabelas, de forma a aproveitar melhor a taxa de comunicação efetiva do driver de comunicação. Por exemplo, cada posição numa tabela de memórias para monitoração de alarmes deve estar associada a 16 TAGs de alarme digital.
- Independente da velocidade do canal de comunicação utilizado, em condições normais e sem o uso da instrução LAI, o CP só responde a uma requisição de comunicação a cada varredura,

assim, é preferível concentrar os dados numa única tabela a realizar uma série de comunicações. Esta consideração é válida apenas para tabelas que respeitam o tamanho limite do bloco de dados, por mensagem, suportados pelo protocolo. Tabelas maiores são sempre transferidas através de duas ou mais requisições feitas pelo driver. Estes limites são apenas para orientação, podendo ser um pouco maiores dependendo do tipo de driver.

- Os operandos devem ser agrupados por utilização. Informações apresentadas numa mesma tela da estação devem ser monitoradas através de uma mesma tabela. Isto torna-se muito significativo no desempenho de estações de supervisão que permitem a declaração de TAGs que são monitorados apenas quando estão presentes na tela sinóptica ativa.
- Deve-se evitar alocar numa mesma tabela variáveis com diferentes freqüências de atualização. Os TAGs de variação lenta devem ser agrupados em tabelas separadas dos TAGs com requisitos de atualização mais exigente. Este princípio vale como critério para quebrar tabelas grandes que já correspondem a mais de uma requisição do driver. Vamos tomar como exemplo uma tabela de 120 posições monitorada a cada segundo que já corresponde a uma quebra com duas requisições do driver por segundo. Se a tabela for redefinida como duas tabelas de 60 posições, onde uma delas for monitorada a cada três segundos, a ocupação do canal de comunicação será reduzido para 1,3 requisições do driver por segundo ou quatro comunicações a cada três segundos.
- Ao contrário dos forçamentos, que são comunicações assíncronas com polling desabilitado, qualquer monitoração definida impõe alguma carga de processamento contínua e contribui para a taxa de utilização do canal de comunicação. Com o recurso das "mensagens não solicitadas" enviados do CP para a estação de supervisão pode-se evitar isto. Apenas quando ocorre algum evento no processo, o CP toma a iniciativa de transmitir uma tabela não solicitada, sendo este tipo de mensagem adequada para dados que não precisam ser adquiridos periodicamente. O uso mais comum deste recurso é na definição de uma tabela de alarmes.
- Quando apenas em determinados momentos, a estação de supervisão precisa adquirir um conjunto de informações, também pode-se utilizar o recurso de mensagens não solicitadas, bastando que o evento que dispara a transmissão seja oriundo de um forçamento enviado pela estação. Para realizar este comando, pode ser alocada a primeira posição de uma tabela não solicitada. Apenas esta posição corresponderia a um TAG de escrita ao contrário dos demais.

Estas sugestões de projeto devem ser aplicadas ao detalhar as entradas e saídas de um sistema de comunicação com o CP, que é uma das primeiras atividades a serem executadas no desenvolvimento de software de CPs. Estas informações podem ser levantadas e registradas num documento denominado Interface de Comunicação e Entradas e Saídas.

Um exemplo de documento deste tipo é a planilha eletrônica associada a esta nota de aplicação: NAP103.XLS. Esta planilha é uma sugestão de documento a ser seguido, devendo ser adaptado para as necessidades do projeto em questão.

A partir deste documento pode-se implementar os bancos de dados de tempo real (BDTR) utilizados pelos sistemas de supervisão. Estes bancos de dados concentram os dados coletados ou enviados do e para o processo através de CPs ou outros equipamentos de campo, informações apresentadas ou digitadas nas telas da estação de supervisão e mensagens trocadas com outros sistemas.

No documento de Interface de Comunicação e Entradas e Saídas devem ser definidos os TAGs correspondentes as interfaces ou mensagens trocadas com os CPs. Como este documento define a maior parte do banco de dados, sua abrangência foi estendida para incluir a totalidade dos TAGs, ou

seja, para incluir até mesmo os TAGs de simulação não utilizados, nem indiretamente, por nenhuma mensagem de comunicação com os CPs.

5.3 Dimensionamento de Timeouts TCP e de Aplicação

A performance de um sistema de comunicação, baseado numa rede Ethernet do tipo TCP, depende, fundamentalmente, das características dos equipamentos envolvidos. Alguns dos parâmetros de comunicação destes equipamentos podem ser ajustados para melhorar o desempenho da rede.

Seguem os principais parâmetros que devem ser observados:

- Timeout da camada de aplicação;
- Timeout da camada de transporte TCP;
- Número de retentativas da camada de transporte TCP;
- Atraso no envio do ACK da camada de transporte TCP.

Estes parâmetros são característicos de todas as interfaces de comunicação Ethernet, seja ela uma interface de um CP, ou uma interface de rede de um microcomputador, bem como das aplicações que acessam os mesmos para comunicação (software executivo ou sistemas de supervisão).

A seguir é apresentado um modelo que tem por objetivo auxiliar na determinação dos melhores valores para estes parâmetros.

Variáveis do sistema a serem consideradas:

- **Tr** - pior tempo de rede para transmissão de qualquer frame do cliente para o servidor, ou do servidor para o cliente. Tratam-se de atrasos inseridos por hubs, switches, roteadores, comprimentos de cabos, e quaisquer outros equipamentos intermediários entre o cliente e o servidor. Para WANs este tempo pode ser bem significativo, enquanto que em LANs, normalmente, é desprezível perante os demais tempos.
- **Tts** - tempo de latência máxima do nível TCP do servidor para processar uma requisição recebida, deixando pronto o ACK TCP desta requisição para transmissão e passando a requisição para o nível de aplicação do servidor.
- **Tas** - tempo de latência máxima do nível de aplicação do servidor para processar uma requisição recebida, deixando pronta uma resposta para o nível TCP transmitir. Por exemplo:
  - CP com AL-3414 é servidor, e foi configurado para responder até 8 requisições a cada ciclo de varredura (instruções LAIs). O tempo máximo de varredura do CP é 100ms.
  - Existem 6 clientes conectados a este servidor, mas dois deles podem gerar, no máximo, até 4 requisições simultâneas para este CP. Desta forma, o servidor pode receber até 12 requisições simultâneas (2 * 4 + 4 * 1).
  - Portanto, neste exemplo, o CP pode demorar até 2 ciclos de varredura para responder às 12 requisições, já que só pode responder a 8 requisições por varredura. Portanto, **Tas** = 200ms para este exemplo.
- **Das** - atraso proposital do servidor antes de transmitir o ACK TCP da requisição. Caso **Das** seja maior que **Tas**, o ACK TCP da requisição e a resposta de aplicação são transmitidos no mesmo frame para o cliente.

- **Ttc** - tempo de latência máxima do nível TCP do cliente para processar uma resposta recebida de um servidor, deixando pronto o ACK TCP desta resposta para transmissão, e passando a resposta para o nível de aplicação do cliente.
- **Dac** - atraso proposital do cliente antes de transmitir o ACK TCP da resposta, com o objetivo de tentar juntar no mesmo frame este ACK TCP com uma nova requisição para o mesmo servidor (se houver alguma agendada).

As variáveis **Das** e **Dac** correspondem ao parâmetro ATRASO DE ENVIO DE ACK DO TCP. Uma das opções é manter este parâmetro nulo, provocando o envio imediato dos ACKs TCP. A outra é inserir estes atrasos propositais. Neste caso, o dimensionamento ideal deste parâmetro, em modo servidor, é algo um pouco maior do que **Tas** (por exemplo: 1.1 * **Tas**), para que o nível de aplicação já tenha a resposta pronta para transmitir junto com o ACK TCP da requisição. No modo cliente, isto corresponderia a um valor pouco maior do que o tempo necessário para o nível de aplicação preparar nova mensagem, ou seja, um valor similar a **Tas**. Portanto, pode-se utilizar a equação “1.1 * **Tas**” para ambos os casos.

Outro parâmetro que pode ser facilmente dimensionado é o NÚMERO DE RETENTATIVAS DO TCP. Este parâmetro pode auxiliar na recuperação de erros transitórios. Em caso de erros permanentes, como quebra de uma linha de transmissão, sabe-se que a recuperação não é possível, e, portanto, de nada adiantarão as retentativas. Se este parâmetro tiver um valor muito elevado, a confirmação de erros permanentes torna-se muito demorada, degradando o desempenho do sistema. Se for nulo ou muito baixo, poderá ser insuficiente para a recuperação de erros transitórios. Aconselha-se manter o valor default atribuído a este parâmetro, exceto em redes de má qualidade, onde seja necessário aumentá-lo. Considerando que os demais parâmetros estão bem dimensionados, duas retentativas não serão suficientes para recuperar erros transitórios se ocorrerem três erros consecutivos em transmissões do nível TCP, o que é pouco provável.

Consideradas as variáveis definidas anteriormente, e os dimensionamentos já realizados para os parâmetros ATRASO DE ENVIO DE ACK DO TCP e NÚMERO DE RETENTATIVAS DO TCP, deve-se observar diversos casos antes de dimensionar os demais parâmetros:

- no modo servidor, o TIMEOUT INICIAL DO TCP deve ser maior do que “1.5 * (Tr + Ttc + Dac + Tr)”, considerando 50% de margem de segurança. Deve ser considerado o valor máximo desta equação, analisando-se todos os possíveis clientes deste servidor.
- no modo cliente, o TIMEOUT INICIAL DO TCP deve ser maior do que “1.5 * (Tr + Tts + Das + Tr + Ttc)”, considerando 50% de margem de segurança. Deve ser considerado o valor máximo desta equação, analisando-se todos os possíveis servidores deste cliente.
- no modo cliente, o timeout de aplicação:
  - deve ser maior do que “1.5 * TIMEOUT INICIAL DO TCP * ($2^{(\text{NÚMERO DE RETENTATIVAS DO TCP + 1})}$ - 1)”, considerando 50% de margem de segurança;
  - também deve ser maior do que “1.5 * (**Tr** + **Tts** + **Tas** + **Tr** + **Ttc**)”, considerando 50% de margem de segurança;
  - deve ser considerado o valor máximo obtido após análise de todos os servidores do qual a interface é cliente.

ATENÇÃO:

O mal dimensionamento dos parâmetros anteriores (timeout inicial do TCP, número de retentativas do TCP, timeout de aplicação, atraso de envio de ACK do TCP) pode provocar degradações na rede, tais como replicações desnecessárias de mensagens, falhas em cascata, tempo excessivo para recuperação de falhas, entre outras, afetando diretamente o desempenho da rede.

5.4 Timeout Inter Sub-rede das Instruções ECR e LTR

Uma característica importante a ser observada para otimizar a comunicação da interface na rede Ethernet Altus, utilizando o protocolo ALNET II sobre TCP/IP, é o parâmetro de TIMEOUT INTER SUB-REDE, configurado a partir do MasterTool, botão ALNET II... do módulo C000.

Timeout Inter Sub-rede

O tempo definido por este campo corresponde ao timeout das instruções ECR e LTR utilizadas na aplicação do CP, e vale tanto para as redes puramente ALNET II (canal serial ALNET II da UCP AL-2004) quanto para as Ethernet (AL-3405, AL-3412, AL-3414, PO7091 e PO7092).

O TIMEOUT INTER SUB-REDE define o tempo máximo que instruções ECR e LTR vão esperar por uma mensagem de resposta do servidor endereçado por estas instruções. Para dimensionamento deste tempo de timeout, vários aspectos importantes devem ser considerados, como mostrado na seção ***Dimensionamento de Timeouts TCP e de Aplicação***.

5.5 Checklist de Verificação da Configuração

Arquiteturas compostas por elementos que se comunicam via rede Ethernet necessitam ser corretamente configurados para que se obtenha o melhor desempenho das comunicações efetuadas.

A tabela a seguir apresenta alguns itens a serem verificados visando o funcionamento adequado de dispositivos da rede Ethernet, visando sempre o melhor desempenho do sistema.

#	Descrição
1	Verificar se todos os operandos declarados na configuração do driver de comunicação estão declarados no CP – atenção para operandos tabela, que devem possuir, no mínimo, o mesmo número de posições.
2	Verificar se os operandos estão declarados conforme orientação do driver utilizado. Drivers com agrupamento automático o fazem em grupos de 220 bytes (exceções consultar tabela ***Número máximo de operandos por bloco de dados***). Assim, devem ser declarados blocos múltiplos deste valor, visando um melhor desempenho do sistema.
3	Verificar se operandos utilizados somente para escrita possuem varredura cíclica desativada. Os mesmos devem ser feitos apenas quando os seus valores sofrerem alguma modificação ou na inicialização do CP.
4	Dar preferência na utilização de mensagens não solicitadas para blocos de dados cujos valores sofrem modificações esporadicamente (eventos e alarmes).
5	Verificar se as mensagens não solicitadas, quando existirem, estão corretamente configuradas.
6	Verificar se o tempo de timeout e retry estão corretamente configurados para a aplicação. Somente o processo pode determinar quando deve-se definir um tempo de timeout maior, com menor número de retries, ou vice-versa.
7	Separar redes de supervisão e de controle das redes corporativas, através do uso de placas adicionais nos computadores (redes completamente separadas) ou através de firewalls ou roteadores que isolem o trafego de dados broadcasts/multicasts.
8	Verificar se o banco de dados do sistema está configurado com diferentes taxas de requisições, conforme a necessidade do processo, a fim de não coletar informações sem necessidade e onerar a performance.

Checklist de verificação da configuração de sistemas de comunicação por Ethernet

6. Ferramentas de Manutenção

6.1 Comando PING

Ferramenta do sistema operacional para avaliação da rede que interliga dois dispositivos, normalmente um computador e um dispositivo remoto, baseada em mensagens do protocolo ICMP (Internet Control Message Protocol).

O funcionamento do comando é muito simples: o computador transmite um pacote de dados endereçado ao dispositivo remoto (endereço IP), que por sua vez deve responder ao computador com uma imagem do pacote de dados. O computador espera por determinado tempo a resposta do dispositivo remoto, não existindo retentativas de transmissão.

A seguir um exemplo de uso do comando PING:

```
C:\WINDOWS>PING  <endereço IP do módulo Ethernet>
Disparando contra <módulo Ethernet> [<IP do módulo Ethernet>] com 32 bytes de dados:
Resposta de <IP do módulo Ethernet>: bytes=32  tempo=5ms  tempo de vida=255
Resposta de <IP do módulo Ethernet>: bytes=32  tempo=3ms  tempo de vida=255
Resposta de <IP do módulo Ethernet>: bytes=32  tempo=3ms  tempo de vida=255
Resposta de <IP do módulo Ethernet>: bytes=32  tempo=3ms  tempo de vida=255

Estatísticas do Ping para <IP do módulo Ethernet>:
      Pacotes:  Enviados=4,  Recebidos=4,  Perdidos=0 (0% de perda),
Tempos aproximados de ida e volta em milisegundos:
      Mínimo=3ms,  Máximo=5ms,  Média=3ms
```

Depois que o comando é executado, o que pode levar alguns segundos, é apresentada a estatística para o comando, com informações importantes como quantidade de pacotes de dados perdidos e tempos mínimo, máximo e médio de resposta. O tempo pode variar em função da distância entre os equipamentos. O tempo máximo deve ser considerado para ajustar o parâmetro de timeout do protocolo de comunicação TCP/IP dos dispositivos envolvidos.

Alguns dos parâmetros do comando PING podem ser definidos pelo usuário, como por exemplo o tamanho do pacote de dados ou a quantidade de pacotes a serem transmitidos. Estes parâmetros podem ser visualizados através da ajuda ao comando:

```
C:\WINDOWS>PING /h
```

6.2 Comando ARP

Ferramenta do sistema operacional para exibição e modificação das tabelas de conversão do endereço IP para endereços físicos pelo protocolo de resolução de endereços (ARP). A tabela de conversão apresenta os endereços dos equipamentos com quem o sistema operacional tinha conexão estabelecida recentemente.

Uma resposta típica para o comando de visualização da tabela de conversão é a seguinte:

```
C:\WINDOWS>ARP -a
Interface: < IP do sistema operacional> on Interface <número da interface de rede>
Endereço IP                     Endereço físico                 Tipo
<IP do módulo Ethernet 1>       <MAC do módulo Ethernet 1>      <tipo de conexão 1>
<IP do módulo Ethernet 2>       <MAC do módulo Ethernet 2>      <tipo de conexão 2>
...                             ...                             ...
<IP do módulo Ethernet N>       <MAC do módulo Ethernet N>      <tipo de conexão N>
```

No caso de substituição de um módulo Ethernet, ocorre a mudança do endereço físico (MAC) relacionado ao endereço IP do respectivo módulo Ethernet, uma vez que cada dispositivo possui um endereço físico único. Logo após a substituição, o sistema operacional pode permanecer com a tabela de conversão desatualizada, o que impede o estabelecimento de conexões para comunicação de forma imediata.

Uma solução para acelerar a atualização da tabela de conversão do endereço IP para endereços físicos é a eliminação da entrada do respectivo IP do módulo da tabela como mostra o comando a seguir:

```
C:\WINDOWS>ARP -d <IP do módulo Ethernet>
```

Alguns módulos Ethernet, como o AL-3414, são capazes de atualizar automaticamente as tabelas de conversão dos clientes através de um comando de ARP gratuito, no momento em que os mesmos são configurados, sendo dispensável o uso do comando ARP manual. Alguns sistemas de supervisão também são espertos o suficiente para forçar um comando ARP pelo sistema operacional assim que o dispositivo deixa de responder.

6.3 Analisador Ethereal

O Ethereal é um analisador de tráfego de rede (sniffer). Trata-se de um software de livre distribuição, com muitas funcionalidades e características interessantes que facilitam a captura e a análise de mensagens Ethernet.

Esta ferramenta pode ser utilizada para avaliar melhora as características da rede e dos dispositivos interligados a mesma, permitindo a visualização das mensagens e a extração de estatísticas de comunicação.

6.3.1 Características

- Disponível para Unix e Windows;
- Interface GUI - a interface com o usuário é gráfica;
- Formato libpcap - interface com o sistema para captura de mensagens a nível de usuário;
- Aplicação de filtros pré e pós captura de mensagens através de vários critérios;
- Pesquisa de mensagens utilizando filtros, que também podem ser utilizados para destacar e colorir seletivamente a informação do sumário da mensagem;
- Três janelas de visualização das mensagens capturadas: lista com um sumário de cada mensagem; árvore com todas as camadas ISO-OSI da mensagem; conteúdo da mensagem em formato hexadecimal;
- Estatísticas de captura e visualização;
- Salva capturas em vários formatos, permitindo a visualização dos mesmos através de outros programas;
- Captura de dados para arquivos com possibilidade de captura circular, utilizando neste caso sempre os mesmos arquivos.

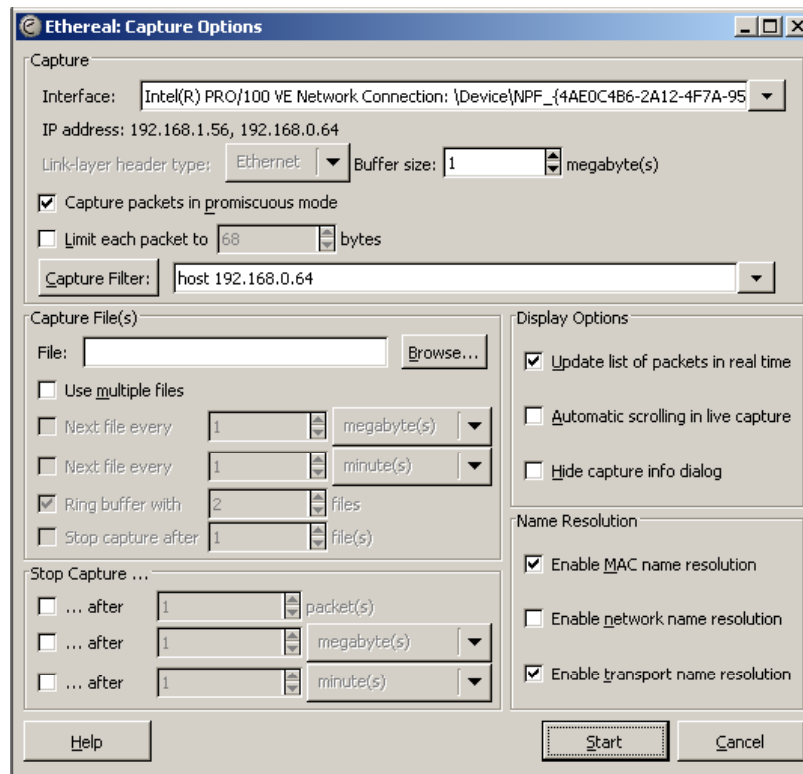
6.3.2 Instalação

São dois os programas que devem ser instalados: o software Ethereal e o WinPcap. Ambos podem ser baixados do site http://www.ethereal.com/.

6.3.3 Captura de Mensagens

Acesse o menu `CAPTURE > OPTIONS...`. Uma janela equivalente a figura a seguir deverá se abrir, permitindo que o usuário faça as configurações necessárias.

ATENÇÃO:

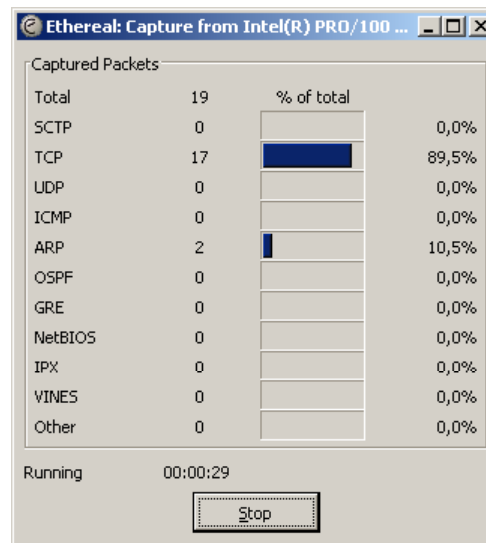
Dependendo da versão do Ethereal, poderão ser observadas algumas pequenas diferenças nas figuras apresentadas a seguir.

Janela com as opções de captura de mensagens

No campo `INTERFACE` deve ser selecionada a interface de rede correta. Podem aparecer várias interfaces, mesmo que o micro só disponha de um cartão de rede. Se o usuário não conhece a placa de rede que está utilizando, será necessário testar todas as opções.

O campo `CAPTURE FILTER` permite selecionar os IPs que serão monitorados (filtro de pré captura). Conforme exemplo apresentado, são monitorados apenas as mensagens do IP 192.168.0.64. Para a seleção de mais IPs, basta insira mais cláusulas HOST separadas por OR (exemplo: host 192.168.0.64 OR host 192.168.0.2).

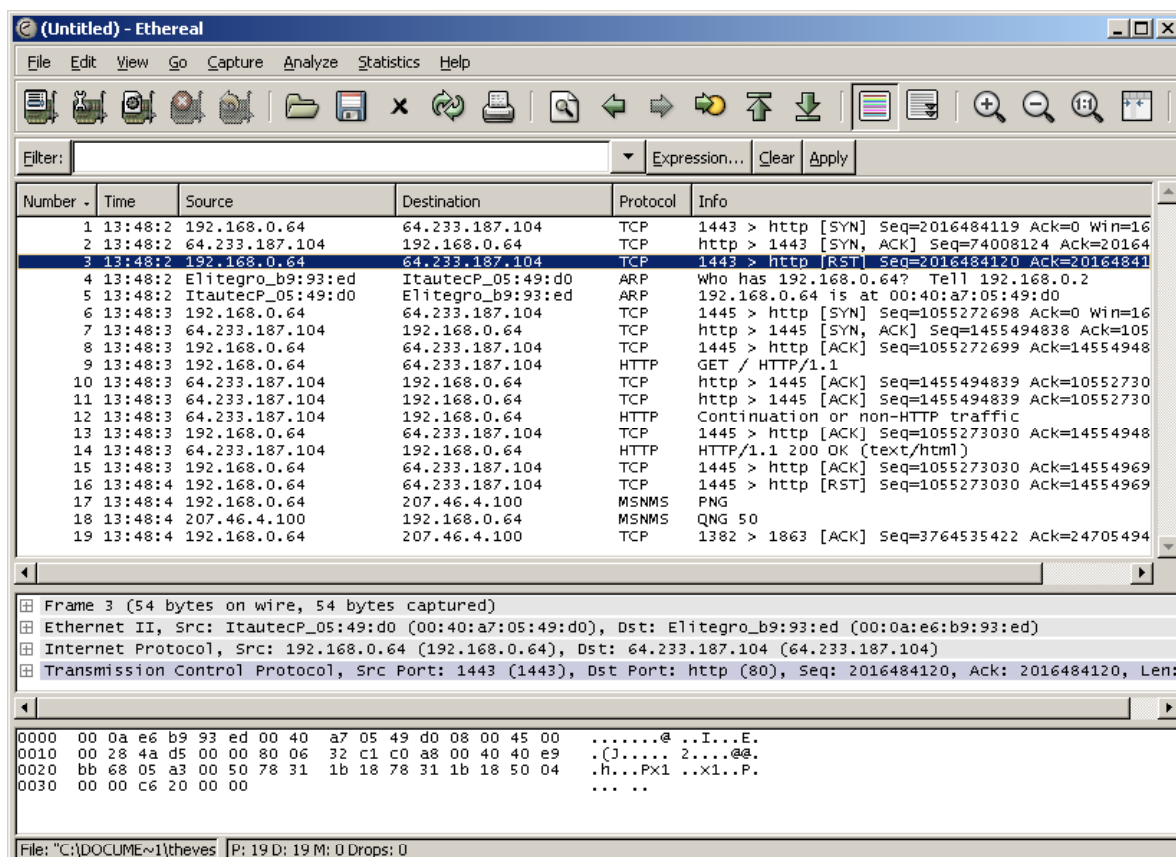
A opção `UPDATE LIST OF PACKETS IN REAL TIME` quando habilitada, faz com que todas as mensagens capturadas e filtradas pelo filtro pós captura sejam apresentadas em tempo real, durante a captura, na janela de visualização das mensagens.

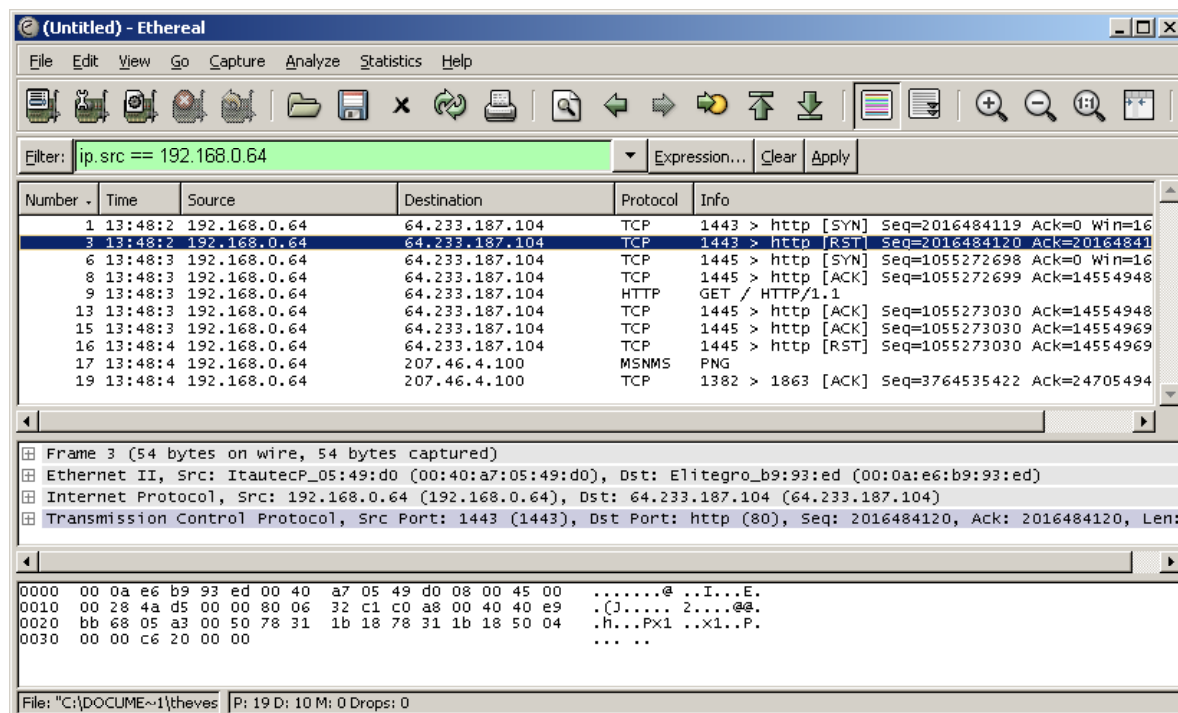
Selecione `START` para começar a monitoração. Neste instante aparece a janela de informação da captura, apresentada na figura a seguir. Se nenhuma mensagem for capturada pode indicar que a interface selecionada é inválida, ou que nenhuma mensagem filtrada está trafegando na rede.

Ao pressionar o botão `STOP`, de parada da captura das mensagens, automaticamente a janela é fechada e todas as mensagens capturadas filtradas são apresentadas na janela principal do Ethereal.

Janela de estatísticas durante a captura das mensagens

Ao pressionar o botão STOP, de parada da captura das mensagens, automaticamente a janela é fechada e todas as mensagens capturadas filtradas são apresentadas na janela principal do Ethereal.

6.3.4 Apresentação das Mensagens

A figura a seguir apresenta a janela principal do Ethereal, com o resultado da captura. As informações mais importantes das mensagens podem ser alinhadas em colunas para facilitar a interpretação.

Janela de apresentação das mensagens capturadas

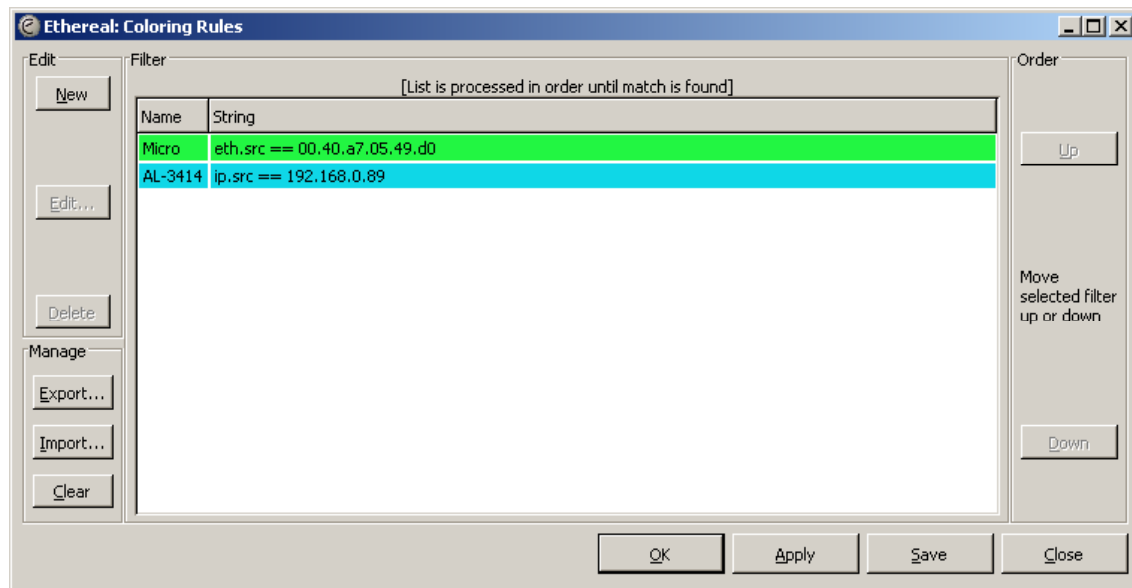
O campo `FILTER` permite definir um filtro pós captura. A figura a seguir apresenta as mensagens filtradas utilizando-se o seguinte filtro: ip.src == 192.168.0.64.

Janela de apresentação das mensagens filtradas

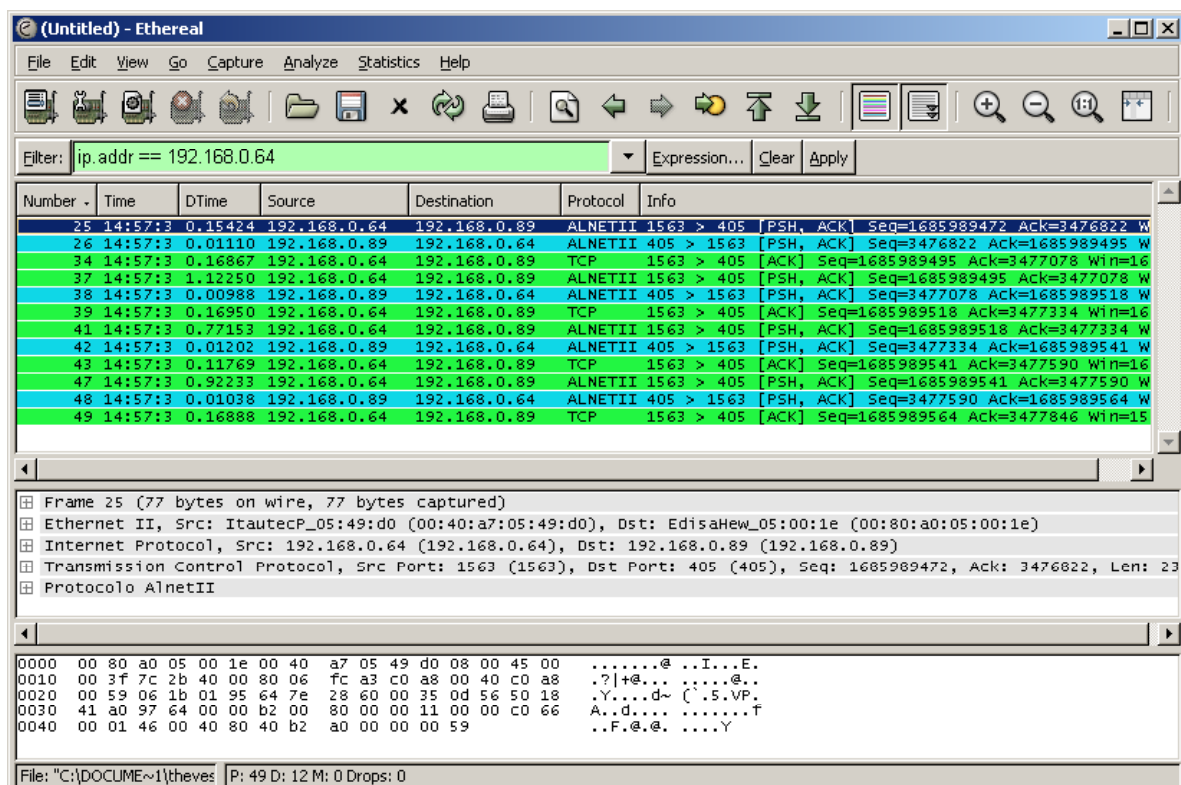
Para facilitar a interpretação das mensagens pelo usuário, pode-se aplicar regras para colorir as mensagens em função de características das mesmas. Por exemplo: apresentar todas as mensagens de um determinado IP com a cor de fundo amarela.

Estas regras podem ser definidas a partir do menu `VIEW > COLORING RULES...`, especificando-se filtros de cores. Cada filtro da lista possui uma ordem de prioridade para ser aplicada e possui quatro parâmetros:

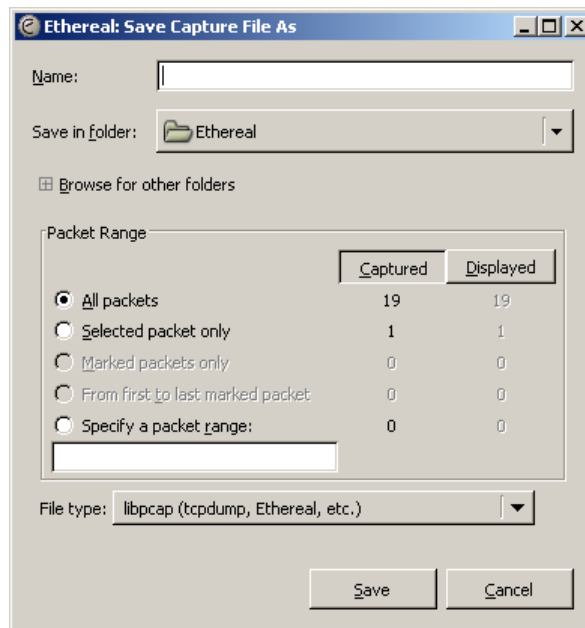
- Name: nome do filtro de cor
- String: filtro a ser aplicado às mensagens capturadas
- Foreground color: cor dos caracteres das mensagens filtradas
- Background color: cor de fundo das mensagens filtradas

As regras de formação do filtro de cor (string) são as mesmas que podem ser empregadas na definição do campo `FILTRO`. Ver figura anterior.

A figura a seguir traz dois exemplos de definição de filtros de cor. O primeiro, nomeado de Micro, irá destacar todas as mensagens originadas do equipamento com endereço MAC 00.40.A7.05.49.D0 com a cor de fundo verde claro. O segundo, nomeado de AL-3414, irá destacar todas as mensagens originadas do equipamento com endereço IP 192.168.0.89 com a cor de fundo azul claro.

Configuração das regras de cor das mensagens

A figura a seguir mostra um exemplo com mensagens filtradas, utilizando-se as regras de cores acima definidas.

Janela de apresentação das mensagens filtradas com regras de cores

6.3.5 Salvando as Mensagens

Após a captura, pode-se salvar todas as mensagens em um arquivo através do menu `FILE > SAVE AS`, conforme a figura abaixo.

Janela de opções para salvar as mensagens capturadas

Deve-se definir o nome do arquivo no campo NAME (utilizar a extensão .ETH para facilitar a identificação do tipo do arquivo e a associação com o Ethereal), a pasta através do botão SAVE IN FOLDER e os tipos de mensagens através dos botões CAPTURED (todas as mensagens capturadas) ou DISPLAYED (somente as mensagens visualizadas a partir do filtro pós captura). Selecione o botão SAVE para gerar o arquivo.

6.3.6 Captura de Mensagens Direto para Arquivos

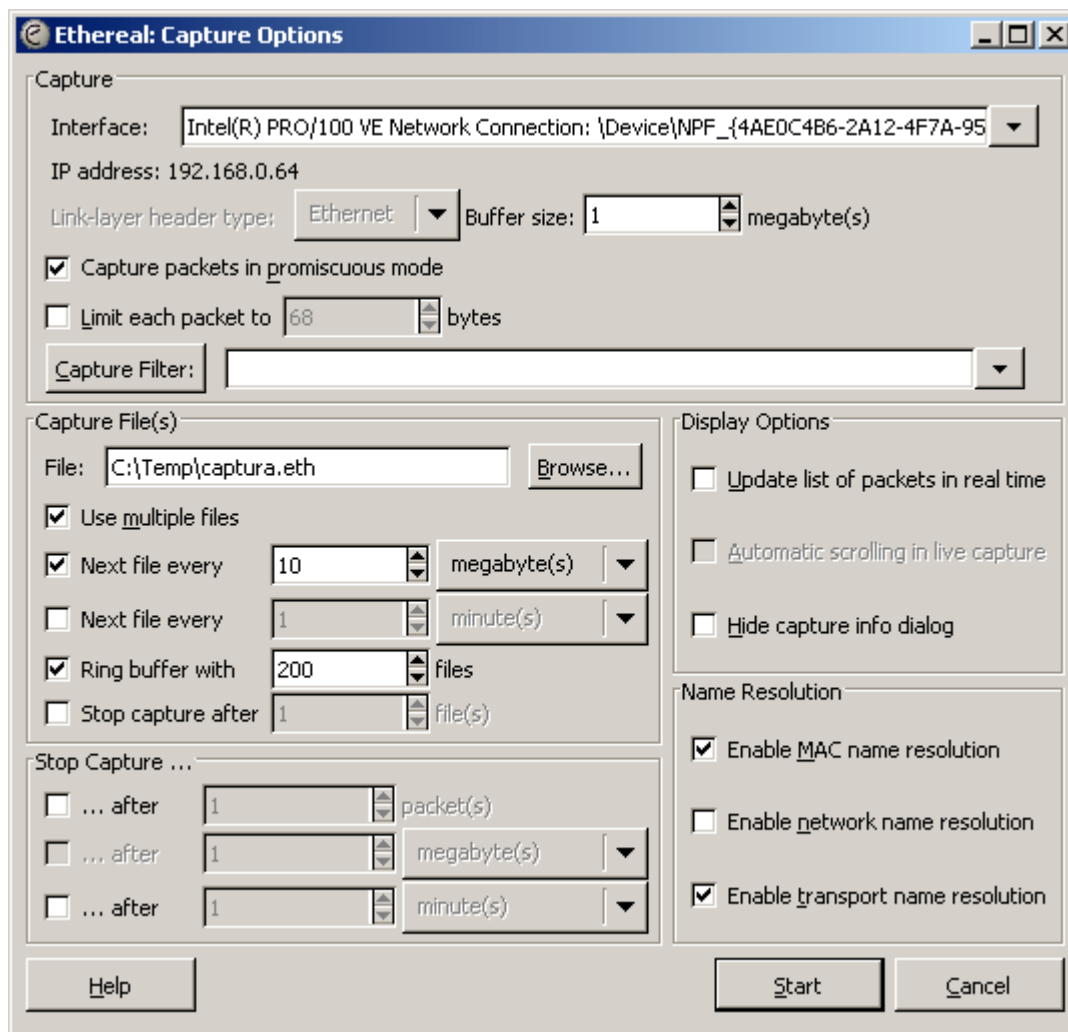
É possível capturar mensagens diretamente para arquivos, durante longos períodos de tempo. A configuração da aplicação Ethereal para operar de tal maneira se dá através da janela com as opções de captura de mensagens CAPTURE > OPTIONS....

As opções de captura de mensagens diretamente para arquivos se dá a partir da célula CAPTURE FILE(S) da janela de configuração das opções. No campo FILE deve ser especificado o caminho e o nome do arquivo que deve ser gerado contendo as mensagens capturadas.

A opção USE MULTIPLE FILES permite a geração de vários arquivos em vez de um único, sendo que a condição de quebra do arquivo pode ocorrer por tamanho ou por tempo: opção NEXT FILE EVERY selecionada para megabytes ou minutos.

A opção RING BUFFER WITH permite especificar a quantidade máxima de arquivos que podem ser gerados. Se este limite for atingido, a aplicação irá começar a sobrescrever sobre os primeiros arquivos, de forma circular.

Cada arquivo gerado traz, além do nome especificado pelo usuário, o número de seqüência do arquivo, seguido da informação de data (ano, mês e dia) e horário (hora, minuto e segundo) da primeira mensagem presente no arquivo. Exemplo de arquivo gerado:

CAPTURA_00001_20070227111449.ETH

Janela de opções para capturar mensagens direto para arquivos

7. Revisões

A revisão deste documento é mostrada na margem superior, indicando alterações no conteúdo ou melhorias no formato.

O histórico a seguir apresenta observações correspondentes a cada revisão:

Revisão: A

Data: 28/02/2007

Autoria: Nelson L. Theves

Aprovação: Júlio A. Sieczkowski

Natureza das alterações:

Primeira emissão do documento.